

# Implementarea filtrelor digitale FIR în forma lattice

Laborator 7, PSS

## Table of contents

1	Obiectiv	1
2	Noțiuni teoretice	1
3	Exerciții teoretice	2
4	Exerciții practice	2
5	Întrebări finale	3

## 1 Obiectiv

Familiarizarea studenților cu formele de implementare tip *lattice* folosite la implementarea filtrelor de tip FIR

## 2 Noțiuni teoretice

Implementarea în formă *lattice* a unui filtru FIR de ordin 3:

Ecuatii:

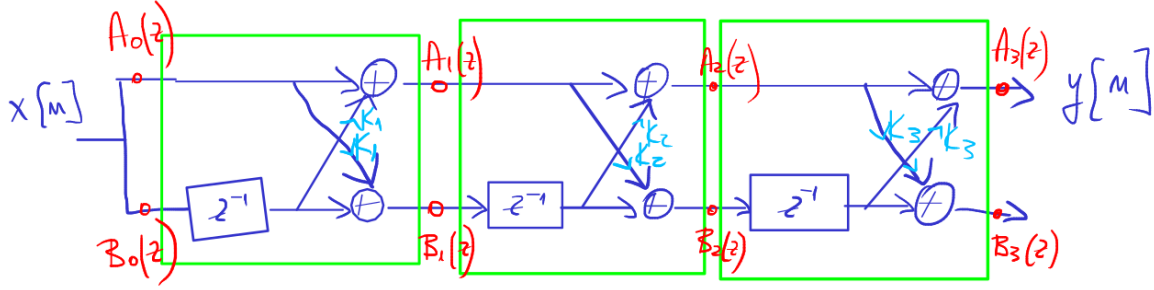


Figure 1: Forma lattice, ordin 3

$$\begin{aligned}
 A_0(z) &= B_0(z) = 1 \\
 A_m(z) &= A_{m-1}(z) + K_m \cdot z^{-1} \cdot B_{m-1}(z) \\
 A_{m-1}(z) &= \frac{A_m(z) - K_m \cdot B_m(z)}{1 - K_m^2} \\
 B_m(z) &= z^{-m} B_m(z^{-1}) = \text{similar cu } A_m(z), \text{ cu coeficienții în ordine inversă}
 \end{aligned}$$

Aceste ecuații permit calcularea coeficienților de reflexie  $K_m$  din  $H(z)$ , sau calcularea  $H(z)$  dacă se cunosc  $K_m$ .

### 3 Exerciții teoretice

1. Determinați coeficienții filtrului FIR în forma directă dacă se cunosc coeficienții de reflexie ai structurii *lattice*:  $K_1 = \frac{1}{2}$ ,  $K_2 = 0.6$ ,  $K_3 = -0.7$ ,  $K_4 = \frac{1}{3}$ .
2. Determinați coeficienții structurii *lattice* pentru un filtru FIR cu funcția de sistem:

$$H(z) = 1 + \frac{2}{5}z^{-1} + \frac{7}{20}z^{-2} + \frac{1}{2}z^{-3}$$

### 4 Exerciții practice

1. În Matlab, utilizați utilitarul `fdatool` pentru a proiecta unul din filtrele următoare:
  - a. Un filtru trece-jos FIR de ordin 5, de tip echiriplu, cu frecvența de tăiere de 5kHz la o frecvență de eșantionare de 44.1kHz;
  - b. Un filtru trece-sus FIR de ordin 5, de tip echiriplu, cu frecvența de tăiere de 1kHz la o frecvență de eșantionare de 44.1kHz;
  - c. Un filtru trece-bandă FIR de ordin 5, de tip echiriplu, cu banda de trecere între 700Hz și 4kHz la o frecvență de eșantionare de 44.1kHz.

2. Exportați coeficienții filtrului de mai sus în Workspace-ul Matlab, cu numele `Num`.

Utilizați funcția `tf2latc()` pentru a converti coeficienții din vectorul `Num` (corespunzători formelor directe) în coeficienți ai formei *lattice*. Denumiți vectorul rezultat  $K$  și afișați-l.

Utilizați apoi și funcția inversă `latc2tf()` pentru a converti coeficienții din forma *lattice*  $K$  înapoi în coeficienți ai formei directe, și verificați că se obțin aceleași valori ca în `Num`.

3. În mediul Simulink, realizați implementarea FIR filtrului de mai sus în forma *lattice*.
  - a. Implementați schema, punând în Gain-uri coeficienții din vectorul  $K$  ( $K(1)$ ,  $K(2)$ , etc)
  - b. Afișați răspunsul la impuls (intrarea “Discrete Impulse”) și răspunsul la treapta unitate (intrarea “Step”).

4. În Matlab, creați o funcție pentru a filtra un vector  $x$  cu un filtru FIR pentru care se cunosc coeficienții *lattice*  $K$ , obținând vectorul de ieșire  $y$ .

Găsiți o metodă de a calcula ieșirea  $y[n]$  la un moment oarecare  $n$ , pe baza schemei. Cu alte cuvinte, dacă avem schema, cum putem scrie niște ecuații sau linii de cod pentru a o implementa?

Puteți utiliza următorul template orientativ:

```
function y = filter_lattice_FIR(K, x)
% Filter the vector x with a lattice FIR filter with coefficients K

ord = length(K);

% Write code here:
...

end
```

5. Testați funcția, apelând-o cu coeficienții  $K$  ai filtrului proiectat anterior, pentru semnalul de intrare  $x$  de forma:

- a. un vector cu un 1 urmat de 19 zerouri (impuls unitate)
- b. un vector cu 20 de 1 (treapta unitate)

Afișați grafic vectorii obținuți la ieșire, cu funcția `stem()`.

## 5 Întrebări finale

1. TBD